Subject: [PATCH ext3/ext4] orphan list check on destroy_inode
Posted by vaverin on Mon, 04 Jun 2007 05:18:55 GMT
View Forum Message <> Reply to Message

Customers claims to ext3-related errors, investigation showed that ext3 orphan list has been corrupted and have the reference to non-ext3 inode. The following debug helps to understand the reasons of this issue.

Signed-off-by: Vasily Averin <vvs@sw.ru>

```
diff --git a/fs/ext3/super.c b/fs/ext3/super.c
index 6e30629..46e2fa6 100644
--- a/fs/ext3/super.c
+++ b/fs/ext3/super.c
@@ -459,6 +459,21 @@ static struct inode *ext3_alloc_inode(struct super_block *sb)

 static void ext3_destroy_inode(struct inode *inode)
 {
+ if (!list_empty(&(EXT3_I(inode)->i_orphan))) {
+  int i, imax;
+  unsigned int *p;
+
+  p = (unsigned int *)EXT3_I(inode);
+  imax = sizeof(struct ext3_inode_info) / sizeof(unsigned int);
+  printk("Inode %p: orphan list check failed!\n", EXT3_I(inode));
+  for (i = 0; i < imax; i++) {
+   if (i && ((i % 8) == 0))
+    printk("\n");
+   printk("%08x ", *p++);
+  }
+  printk("\n");
+  dump_stack();
+ }
  kmem_cache_free(ext3_inode_cachep, EXT3_I(inode));
 }

diff --git a/fs/ext4/super.c b/fs/ext4/super.c
index cb9afdd..f86e0ac 100644
--- a/fs/ext4/super.c
+++ b/fs/ext4/super.c
@@ -510,6 +510,21 @@ static struct inode *ext4_alloc_inode(struct super_block *sb)

 static void ext4_destroy_inode(struct inode *inode)
 {
+ if (!list_empty(&(EXT4_I(inode)->i_orphan))) {
+  int i, imax;
+  unsigned int *p;
+
```

```
+  p = (unsigned int *)EXT4_I(inode);
+  imax = sizeof(struct ext4_inode_info) / sizeof(unsigned int);
+  printk("Inode %p: orphan list check failed!\n", EXT4_I(inode));
+  for (i = 0; i < imax; i++) {
+   if (i && ((i % 8) == 0))
+    printk("\n");
+   printk("%08x ", *p++);
+  }
+  printk("\n");
+  dump_stack();
+ }
   kmem_cache_free(ext4_inode_cachep, EXT4_I(inode));
  }
```